

Scalable Trace-norm Minimization by Subspace Pursuit Proximal Riemannian Gradient

Mingkui Tan^{†*}Shijie Xiao[‡]Junbin Gao^{††}Dong Xu[‡]Anton Van Den Hengel[†] Qinfeng Shi[†]

Abstract

Trace-norm regularization plays a vital role in many learning tasks, such as low-rank matrix recovery (MR), and low-rank representation (LRR). Solving this problem directly can be computationally expensive due to the unknown rank of variables or large-rank singular value decompositions (SVDs). To address this, we propose a proximal Riemannian gradient (PRG) scheme which can efficiently solve trace-norm regularized problems defined on real-algebraic variety $\mathcal{M}_{\leq r}$ of real matrices of rank at most r . Based on PRG, we further present a simple and novel subspace pursuit (SP) paradigm for general trace-norm regularized problems without the explicit rank constraint $\mathcal{M}_{\leq r}$. The proposed paradigm is very scalable by avoiding large-rank SVDs. Empirical studies on several tasks, such as matrix completion and LRR based subspace clustering, demonstrate the superiority of the proposed paradigms over existing methods.

1 Introduction

Trace-norm regularization plays a vital role in various areas, such as machine learning [49, 4], data mining [37], computer vision and image processing [26, 34, 46]. Most trace-norm based problems can be formulated into the following general formulation [25]:

$$\min_{\mathbf{X}, \mathbf{E}} \|\mathbf{X}\|_* + \lambda \Upsilon(\mathbf{E}), \text{ s.t. } \mathcal{A}(\mathbf{X}) + \mathcal{B}(\mathbf{E}) = \mathbf{D}, \quad (1)$$

where λ is a regularization parameter, $\|\mathbf{X}\|_*$ is the trace-norm (also known as the nuclear-norm) of a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$, both \mathcal{A} and \mathcal{B} are linear operators depending on

specific applications [25], \mathbf{D} denotes data or observations, \mathbf{E} can be considered as an error term and $\Upsilon(\mathbf{E})$ is a regularizer on \mathbf{E} which is possibly non-smooth. The trace norm $\|\mathbf{X}\|_*$ is the tightest convex lower bound to the rank function $\text{rank}(\mathbf{X})$ [35], and the minimization of (1) encourages the variable \mathbf{X} to be low-rank [14, 15, 9]. Among various trace-norm based problems, the low-rank matrix recovery (MR) [8], and low-rank representation (LRR) [26], have gained particular interest in the last decade.

MR [8] seeks to recover a low-rank matrix \mathbf{X} from partial observations that are recorded in a vector $\mathbf{d} \in \mathbb{R}^l$, where $l \ll mn$. If there are no outliers in the observations, one can recover \mathbf{X} with high probability by solving the following problem [8, 21]:

$$\min_{\mathbf{X}} \|\mathbf{X}\|_*, \text{ s.t. } \mathcal{A}(\mathbf{X}) = \mathbf{d}, \quad (2)$$

which can be deemed as a simplified version of formulation (1). MR has been successfully applied in many tasks such as matrix completion filtering [37, 36]. However, the recovery performance by solving problem (2) might be seriously degraded if the observations contain severe outliers [11, 10]. To improve the robustness, we may introduce an additional variable \mathbf{E} into the constraint $\mathcal{A}(\mathbf{X}) = \mathbf{d}$ as in (1), and regularize it using ℓ_1 -norm regularization (i.e., $\|\mathbf{E}\|_1$) or $\ell_{2,1}$ -norm regularization (i.e., $\|\mathbf{E}\|_{2,1}$) [11, 10].

LRR [26] seeks to find a low-rank representation $\mathbf{X} \in \mathbb{R}^{n \times n}$ of given data $\mathbf{D} \in \mathbb{R}^{m \times n}$ by solving an optimization problem of the following form: $\min_{\mathbf{X}, \mathbf{E}} \lambda \|\mathbf{X}\|_* + \|\mathbf{E}\|_{2,1}$ s.t. $\mathbf{D}\mathbf{X} + \mathbf{E} = \mathbf{D}$, where \mathbf{D} denotes the given data with n samples, and $\|\mathbf{E}\|_{2,1}$ encourages the representation error \mathbf{E} to be column-wise sparse. LRR has been widely applied in many real-world tasks such as motion segmentation and face clustering [26, 25, 45].

Many algorithms have been proposed to solve trace-norm regularized problems [15, 23, 18, 24, 43, 50, 7, 41, 39], but most focus on solving problem (2), such as the singular value thresholding (SVT) [7], augmented Lagrangian method (ALM) and alternating direction method (ADM) [24, 41, 39]. Unlike these methods, some re-

[†]School of Computer Science, The University of Adelaide;
[‡]School of Computer Engineering, Nanyang Technological University; ^{††}Computer Science in the School of Computing and Mathematics, Charles Sturt University.

searchers proposed to solve an equivalent problem to problem (2) [43, 19]: $\min_{\mathbf{X}} \|\mathbf{X}\|_* + \frac{\gamma}{2} \|\mathcal{A}(\mathbf{X}) - \mathbf{d}\|_2^2$, where γ is a regularization parameter. This problem is known as the *matrix lasso*, and can be addressed by proximal gradient (PG) or accelerated proximal gradient (APG) [12, 43, 19].

The optimization of problem (1) is more challenging due to the additional variable \mathbf{E} . By minimizing \mathbf{X} and \mathbf{E} alternatively, the aforementioned methods (*e.g.*, ADM and PG) have been extended to solve this problem [24, 25].

The above methods have shown great success in practice [43, 24]. However, the optimization usually involves repetitive SVDs due to the SVT operation, making them inefficient on large-scale problems [29, 44]. Using homotopy strategies and applying rank prediction techniques may accelerate the convergence speed with truncated SVDs [43, 24, 25]. However, the rank prediction could be non-trivial in general, and large-rank SVDs is still inevitable if the optimal solution has a large rank.

To develop more scalable algorithms, some researchers have tackled a version of the problem in which it is assumed that the rank of \mathbf{X} is known, *e.g.*, $\text{rank}(\mathbf{X}) = r$, and thus proposed to solve a variational form of problem (2) [35, 18]: $\min_{\mathbf{G}, \mathbf{H}} \|\mathbf{G}\|_F^2 + \|\mathbf{H}\|_F^2$, s.t. $\mathcal{A}(\mathbf{G}\mathbf{H}^\top) = \mathbf{D}$, where $\mathbf{G} \in \mathbb{R}^{m \times r}$ and $\mathbf{H} \in \mathbb{R}^{n \times r}$. Many methods have been developed to address this problem, such as gradient based methods [35, 18] and stochastic gradient methods [47, 36, 5]. However, these methods may still suffer from slow convergence speeds [30, 44].

Recently, fixed-rank methods by exploiting the smooth geometry of matrices on fixed-rank manifolds have shown great advantages in computation for solving matrix recovery problems [29, 6, 1], such as the low-rank geometric conjugate gradient method (LRGeomCG) [44], the quotient geometric matrix completion method (qGeomMC) [30], and the method of scaled gradients on Grassmann manifolds for matrix completion (ScGrassMC) [33]. However, these methods can only deal with smooth objectives. Moreover, the rank parameter r is usually unknown in practice, and nontrivial to discover.

Motivated by the superiority of Riemannian gradient-based methods on low-rank matrix recovery problems [44], in this paper, we exploit classical proximal gradient methods and geometries of the real-algebraic variety $\mathcal{M}_{\leq r}$ to address the problem in (1). The main contributions of this paper are as follows:

- We propose a proximal Riemannian gradient (PRG) scheme to address trace-norm regularized problems with explicit rank constraint $\text{rank}(\mathbf{X}) \leq r$. By exploiting geometries on $\mathcal{M}_{\leq r}$, PRG avoids repetitive large-scale SVDs of classical proximal methods, making it more scalable.
- To address general trace-norm regularized problems in (1), we present a simple and novel active subspace

framework which incorporates PRG as a slave solver. This framework does not require the prior knowledge of r and large-rank SVDs, and can even accelerate the convergence speed of PRG.

2 Notations and Preliminaries

Let the superscript \top denote the transpose of a vector/matrix, $\mathbf{0}$ be a vector/matrix with all zeros, $\text{diag}(\mathbf{v})$ be a diagonal matrix with diagonal elements equal to \mathbf{v} , $\langle \mathbf{A}, \mathbf{B} \rangle = \text{tr}(\mathbf{A}\mathbf{B}^\top)$ be the inner product of \mathbf{A} and \mathbf{B} , and $\|\mathbf{v}\|_p$ be the ℓ_p -norm of a vector \mathbf{v} . Let \mathcal{A} be a linear operator with \mathcal{A}^* being its adjoint operator. The operator $\max(\boldsymbol{\sigma}, \mathbf{v})$ operates on each dimension of $\boldsymbol{\sigma}$. Let $\mathbf{X} = \mathbf{U}\text{diag}(\boldsymbol{\sigma})\mathbf{V}^\top$ be the SVD of $\mathbf{X} \in \mathbb{R}^{m \times n}$. The nuclear norm of \mathbf{X} is defined as $\|\mathbf{X}\|_* = \|\boldsymbol{\sigma}\|_1 = \sum_i |\sigma_i|$ and the Frobenius norm of \mathbf{X} is defined as $\|\mathbf{X}\|_F = \|\boldsymbol{\sigma}\|_2$. Lastly, for any convex function $\Omega(\mathbf{X})$, let $\partial\Omega(\mathbf{X})$ denote its subdifferential at \mathbf{X} .

We now introduce some of the basic notions of the geometry of fixed-rank matrices and matrix varieties as follows.

Geometries of Fixed-rank Matrices. The fixed rank- r matrices lie on a smooth submanifold defined below

$$\begin{aligned} \mathcal{M}_r &= \{\mathbf{X} \in \mathbb{R}^{m \times n} : \text{rank}(\mathbf{X}) = r\} \\ &= \{\mathbf{U}\text{diag}(\boldsymbol{\sigma})\mathbf{V}^\top : \mathbf{U} \in \text{St}_r^m, \mathbf{V} \in \text{St}_r^n, \|\boldsymbol{\sigma}\|_0 = r\}, \end{aligned}$$

where $\text{St}_r^m = \{\mathbf{U} \in \mathbb{R}^{m \times r} : \mathbf{U}^\top \mathbf{U} = \mathbf{I}\}$ denotes the Stiefel manifold of $m \times r$ real and orthonormal matrices, and the entries in $\boldsymbol{\sigma}$ are in descending order [44]. Moreover, the tangent space $T_{\mathbf{X}}\mathcal{M}_r$ at \mathbf{X} is given by

$$\begin{aligned} T_{\mathbf{X}}\mathcal{M}_r &= \{\mathbf{U}\mathbf{M}\mathbf{V}^\top + \mathbf{U}_p\mathbf{V}^\top + \mathbf{U}\mathbf{V}_p^\top : \mathbf{M} \in \mathbb{R}^{r \times r}, \mathbf{U}_p \in \mathbb{R}^{m \times r}, \\ &\quad \mathbf{U}_p^\top \mathbf{U} = \mathbf{0}, \mathbf{V}_p \in \mathbb{R}^{n \times r}, \mathbf{V}_p^\top \mathbf{V} = \mathbf{0}\}. \end{aligned} \quad (3)$$

Given $\mathbf{X} \in \mathcal{M}_r$ and $\mathbf{A}, \mathbf{B} \in T_{\mathbf{X}}\mathcal{M}_r$, by defining a metric $g_{\mathbf{X}}(\mathbf{A}, \mathbf{B}) = \langle \mathbf{A}, \mathbf{B} \rangle$, \mathcal{M}_r is a **Riemannian manifold** by restricting $\langle \mathbf{A}, \mathbf{B} \rangle$ to the *tangent bundle* [2].¹ The norm of a tangent vector $\boldsymbol{\zeta}_{\mathbf{X}} \in T_{\mathbf{X}}\mathcal{M}_r$ evaluated at \mathbf{X} is defined as $\|\boldsymbol{\zeta}_{\mathbf{X}}\| = \sqrt{\langle \boldsymbol{\zeta}_{\mathbf{X}}, \boldsymbol{\zeta}_{\mathbf{X}} \rangle}$.

Once the metric is fixed, the notion of the gradient of an objective function can be introduced. For a Riemannian manifold, the **Riemannian gradient** of a smooth function $f : \mathcal{M}_r \rightarrow \mathbb{R}$ at $\mathbf{X} \in \mathcal{M}_r$ is defined as the unique tangent vector $\text{grad}f(\mathbf{X})$ in $T_{\mathbf{X}}\mathcal{M}_r$, such that $\langle \text{grad}f(\mathbf{X}), \boldsymbol{\xi} \rangle = \text{D}f(\mathbf{X})[\boldsymbol{\xi}]$, $\forall \boldsymbol{\xi} \in T_{\mathbf{X}}\mathcal{M}_r$. As \mathcal{M}_r is embedded in $\mathbb{R}^{m \times n}$, the Riemannian gradient of f is given as the **orthogonal projection** of the gradient of f onto the tangent space. Here, the orthogonal projection of any $\mathbf{Z} \in \mathbb{R}^{m \times n}$ onto the tangent space $T_{\mathbf{X}}\mathcal{M}_r$ at $\mathbf{X} = \mathbf{U}\text{diag}(\boldsymbol{\sigma})\mathbf{V}^\top$ is defined as

$$P_{T_{\mathbf{X}}\mathcal{M}_r}(\mathbf{Z}) : \mathbf{Z} \mapsto P_U \mathbf{Z} P_V + P_U^\perp \mathbf{Z} P_V + P_U \mathbf{Z} P_V^\perp. \quad (4)$$

¹The *tangent bundle* is defined as the disjoint union of all tangent spaces $T\mathcal{M}_r = \bigcup_{\mathbf{X} \in \mathcal{M}_r} \{\mathbf{X}\} \times T_{\mathbf{X}}\mathcal{M}_r$.

where $P_U = \mathbf{U}\mathbf{U}^\top$ and $P_U^\perp = \mathbf{I} - \mathbf{U}\mathbf{U}^\top$. Moreover, define $P_{T_0\mathcal{M}_{\leq r}}(\mathbf{Z}) = \mathbf{0}$ when $\mathbf{X} = \mathbf{0}$ [42]. Letting $\mathbf{G} = \nabla f(\mathbf{X})$ be the gradient of $f(\mathbf{X})$ on vector space, it follows that

$$\text{grad}f(\mathbf{X}) = P_{T_{\mathbf{X}}\mathcal{M}_r}(\mathbf{G}). \quad (5)$$

The *Retraction* mapping on \mathcal{M}_r relates an element in the tangent space to a corresponding point on the manifold. One of the issues associated with such retraction mappings is to find the best rank- r approximation to $\mathbf{X} + \boldsymbol{\xi}$ in terms of the Frobenius norm

$$\begin{aligned} R_{\mathbf{X}}(\boldsymbol{\xi}) &= P_{\mathcal{M}_r}(\mathbf{X} + \boldsymbol{\xi}) \\ &= \arg \min_{\mathbf{Y} \in \mathcal{M}_r} \|\mathbf{Y} - (\mathbf{X} + \boldsymbol{\xi})\|_F. \end{aligned} \quad (6)$$

In general, this problem can be addressed by performing SVD on $\mathbf{X} + \boldsymbol{\xi}$, which may be computationally expensive.

Remark 1. Since $\boldsymbol{\xi} = \mathbf{U}\mathbf{M}\mathbf{V}^\top + \mathbf{U}_p\mathbf{V}_p^\top + \mathbf{U}\mathbf{V}_p^\top \in T_{\mathbf{X}}\mathcal{M}_r$, $R_{\mathbf{X}}(\boldsymbol{\xi})$ can be efficiently computed as in Algorithm 6 in [44] with efficient QR decompositions on low rank matrices \mathbf{U}_p and \mathbf{V}_p . The corresponding time complexity is $14(m+n)r^2 + C_{\text{SVD}}r^3$, where $r \ll \min(m, n)$ and C_{SVD} is a moderate constant [44].

Varieties of Low-rank Matrices. Note that the submanifold \mathcal{M}_r is open, and the manifold properties break down at the boundary where $\text{rank}(\mathbf{X}) < r$, and the convergence analysis on \mathcal{M}_r will be difficult accordingly [38]. Therefore, it would be more convenient to consider the closure of \mathcal{M}_r :

$$\mathcal{M}_{\leq r} = \{\mathbf{X} \in \mathbb{R}^{m \times n} : \text{rank}(\mathbf{X}) \leq r\}, \quad (7)$$

which is a real-algebraic variety [38]. Let $\text{ran}(\mathbf{X})$ be the column space of \mathbf{X} . In the singular points where $\text{rank}(\mathbf{X}) = s < r$, we will construct search directions in the tangent cone [38] (instead of the tangent space)

$$T_{\mathbf{X}}\mathcal{M}_{\leq r} = T_{\mathbf{X}}\mathcal{M}_s \oplus \{\boldsymbol{\Xi}_{r-s} \in \mathcal{U}^\perp \otimes \mathcal{V}^\perp\}, \quad (8)$$

where $\mathcal{U} = \text{ran}(\mathbf{X})$ and $\mathcal{V} = \text{ran}(\mathbf{X}^\top)$. Essentially, $\boldsymbol{\Xi}_{r-s}$ is a best rank- $(r-s)$ approximation of $\mathbf{G} - P_{T_{\mathbf{X}}\mathcal{M}_s}(\mathbf{G})$, which can be cheaply computed with truncated SVD of rank $(r-s)$. Let $\text{grad}f(\mathbf{X}) \in T_{\mathbf{X}}\mathcal{M}_{\leq r}$ be the projection of \mathbf{G} on $T_{\mathbf{X}}\mathcal{M}_{\leq r}$. It can be computed by

$$\text{grad}f(\mathbf{X}) = P_{T_{\mathbf{X}}\mathcal{M}_s}(\mathbf{G}) + \boldsymbol{\Xi}_{r-s}. \quad (9)$$

Given a search direction $\boldsymbol{\xi} \in T_{\mathbf{X}}\mathcal{M}_{\leq r}$, we need perform retraction which finds the best approximation by a matrix of rank at most r as measured in terms of the Frobenius norm, i.e.,

$$R_{\mathbf{X}}(\boldsymbol{\xi}) = \arg \min_{\mathbf{Y} \in \mathcal{M}_{\leq r}} \|\mathbf{Y} - (\mathbf{X} + \boldsymbol{\xi})\|_F. \quad (10)$$

Since $\boldsymbol{\Xi}_{r-s} \in \mathcal{U}^\perp \otimes \mathcal{V}^\perp$, $R_{\mathbf{X}}(\boldsymbol{\xi})$ w.r.t. $\mathcal{M}_{\leq r}$ can be efficiently computed with the same complexity as on \mathcal{M}_r (see details in the supplementary file).

3 Proximal Riemannian Gradient on $\mathcal{M}_{\leq r}$

Directly solving the general trace-norm regularized problem in (1) can be computationally expensive due to the unknown rank of variables (regarding fixed-rank methods) or large-rank singular value decompositions (SVDs) (regarding proximal gradient based methods). To make the problem simpler, let us first consider the following problem with explicit rank constraint $\text{rank}(\mathbf{X}) \leq r$

$$\begin{aligned} \min_{\mathbf{X}, \mathbf{E}} \quad & \|\mathbf{X}\|_* + \lambda \Upsilon(\mathbf{E}), \\ \text{s.t.} \quad & \mathcal{A}(\mathbf{X}) + \mathcal{B}(\mathbf{E}) = \mathbf{D}, \mathbf{X} \in \mathcal{M}_{\leq r}. \end{aligned} \quad (11)$$

Here, the parameter r is supposed to be known. Nevertheless, based on (11), we will propose a subspace pursuit paradigm to solve the general trace-norm regularized problem in (1); see details in Section 4.

The penalty method is adopted to deal with the equality constraint $\mathcal{A}(\mathbf{X}) + \mathcal{B}(\mathbf{E}) = \mathbf{D}$ in (11), and it minimizes a penalized function over $\mathcal{M}_{\leq r}$ in the following form²:

$$\Psi(\mathbf{X}, \mathbf{E}) = \|\mathbf{X}\|_* + \lambda \Upsilon(\mathbf{E}) + \frac{\gamma}{2} \|\mathcal{A}(\mathbf{X}) + \mathcal{B}(\mathbf{E}) - \mathbf{D}\|_F^2, \quad (12)$$

where γ is a penalty parameter. Note that when there are no outliers, we can let $\mathcal{B}(\mathbf{E}) = \mathbf{0}$ and $\Upsilon(\mathbf{E}) = 0$, and the objective function $\Psi(\mathbf{X}, \mathbf{E})$ is reduced to

$$\Psi(\mathbf{X}) = \|\mathbf{X}\|_* + \frac{\gamma}{2} \|\mathcal{A}(\mathbf{X}) - \mathbf{D}\|_F^2, \quad \mathbf{X} \in \mathcal{M}_{\leq r}. \quad (13)$$

$\Psi(\mathbf{X})$ is also the objective function of the *matrix lasso* problem [43, 19], so one can adapt classical proximal methods [43, 24] to address it. However, proximal gradient methods which directly operate on vector spaces could be very expensive if large-rank SVDs are required.

In this section, we extend classical proximal methods on vector space [32, 43, 48], and propose a *proximal Riemannian gradient* scheme to minimize (12) and (13) by exploiting geometries over the matrix variety $\mathcal{M}_{\leq r}$.³

3.1 PRG on $\mathcal{M}_{\leq r}$ for Non-outlier Cases

The objective function $\Psi(\mathbf{X})$ regarding non-outlier cases is much simpler than $\Psi(\mathbf{X}, \mathbf{E})$. When there are no outliers, we solve the following optimization problem:

$$\min_{\mathbf{X}} \quad \|\mathbf{X}\|_* + f(\mathbf{X}), \quad \text{s.t. } \text{rank}(\mathbf{X}) \leq r, \quad (14)$$

where $f(\mathbf{X})$ is any smoothing function, for example $f(\mathbf{X}) = \frac{\gamma}{2} \|\mathcal{A}(\mathbf{X}) - \mathbf{D}\|_F^2$.

To introduce proximal methods on $\mathcal{M}_{\leq r}$, similarly as in [43, 24], we introduce a local model of $\Psi(\mathbf{X})$ on $\mathcal{M}_{\leq r}$

²If \mathbf{D} is a vector, the F -norm will be replaced by the ℓ_2 -norm.

³Recall that $\mathcal{M}_{\leq r}$ is a closure of the Riemannian submanifold \mathcal{M}_r . Here, we abuse ‘‘Riemannian’’ for simplicity.

around $\mathbf{Y} \in \mathcal{M}_{\leq r}$ but keeping $\|\mathbf{X}\|_*$ intact:

$$m_L(\mathbf{Y}; \mathbf{X}) := \|\mathbf{X}\|_* + f(\mathbf{Y}) + \langle \text{grad}f(\mathbf{Y}), \boldsymbol{\xi} \rangle + \frac{L}{2} \langle \boldsymbol{\xi}, \boldsymbol{\xi} \rangle,$$

where $\boldsymbol{\xi} \in T_{\mathbf{Y}}\mathcal{M}_{\leq r}$ and $\mathbf{X} = \mathbf{Y} + \boldsymbol{\xi}$. Note the above local model is different from that on vector spaces (see [32, 43]) in the sense that $\mathbf{X} - \mathbf{Y} = \boldsymbol{\xi}$ is restricted on $T_{\mathbf{Y}}\mathcal{M}_{\leq r}$.

Similar to classical proximal gradient methods [32, 43], our proximal Riemannian gradient method solves problem (14) by minimizing $m_L(\mathbf{Y}; \mathbf{X})$ on $\mathcal{M}_{\leq r}$ iteratively. In other words, given $\mathbf{Y} = \mathbf{X}_k$ in the k th iteration, we need to solve the following optimization problem to obtain \mathbf{X}_{k+1} :

$$\min_{\mathbf{X}} m_L(\mathbf{Y}; \mathbf{X}), \text{ s.t. } \mathbf{X} \in \mathcal{M}_{\leq r}. \quad (15)$$

For convenience, let

$$T_L(\mathbf{Y}) := \arg \min_{\mathbf{X} \in \mathcal{M}_{\leq r}} m_L(\mathbf{Y}; \mathbf{X}) \quad (16)$$

be a minimizer of (15). Then it can be computed as follows.

Lemma 1. *Let $\boldsymbol{\xi} = -\text{grad}f(\mathbf{Y})/L$. Denoting the SVD of $R_{\mathbf{Y}}(\boldsymbol{\xi})$ as $R_{\mathbf{Y}}(\boldsymbol{\xi}) = \mathbf{U}_+ \text{diag}(\boldsymbol{\sigma}_+) \mathbf{V}_+^\top$, it follows that $T_L(\mathbf{Y}) = \mathbf{U}_+ \text{diag}(\max(\boldsymbol{\sigma}_+ - 1/L, 0)) \mathbf{V}_+^\top$.*

Please find the proof in supplementary file.

Remark 2. $T_L(\mathbf{Y})$ can be efficiently computed in the sense that $R_{\mathbf{Y}}(\boldsymbol{\xi}) = \mathbf{U}_+ \text{diag}(\boldsymbol{\sigma}_+) \mathbf{V}_+^\top$ can be cheaply computed without expensive SVDs.

Algorithm 1 Proximal Riemannian Gradient for Solving Problem (14).

Require: \mathbf{X}_0 , penalty parameter γ , parameter r , stopping tolerance ϵ .

- 1: For $k = 1, \dots, K$
 - 2: Compute $\text{grad}f(\mathbf{X}_{k-1})$ according to (5) or (9).
 - 3: Choose L_k to satisfy (17), and set $\mathbf{X}_k = T_{L_k}(\mathbf{X}_{k-1})$.
 - 4: Terminate if stopping conditions are achieved.
 - 5: End
 - 6: Return \mathbf{X}_k .
-

PRG iteratively minimizes a local model of Ψ on $\mathcal{M}_{\leq r}$, as shown in Algorithm 1. PRG consists of two major steps 1) compute a search direction in Step 3, and 2) update \mathbf{X}_k according to $\mathbf{X}_k = T_{L_k}(\mathbf{X}_{k-1})$. Here, $1/L_k$ can be deemed as the step size, and it can be determined using Armijo line search. Specifically, given a descent direction $\boldsymbol{\zeta}_k \in T_{\mathbf{X}_k}\mathcal{M}_{\leq r}$, L_k is determined such that

$$\Psi(T_{L_k}(\mathbf{X}_k)) \leq \Psi(\mathbf{X}_k) + \beta \langle \text{grad}f(\mathbf{X}_k), \boldsymbol{\zeta}_k \rangle / L_k, \quad (17)$$

where $\beta \in (0, 1)$.

Optimality condition of (14). A point $\mathbf{X}^* \in \mathcal{M}_{\leq r}$ is a local minimizer of (14) if and only if there exists $\boldsymbol{\varsigma} \in \partial\|\mathbf{X}\|_*$ such that [31]

$$\text{grad}f(\mathbf{X}) + \boldsymbol{\varsigma} = \mathbf{0}. \quad (18)$$

The following lemma guarantees the existence of L_k .

Lemma 2. *Let $\mathbf{X}_k \in \mathcal{M}_{\leq r}$, and $\boldsymbol{\zeta}_k \in T_{\mathbf{X}_k}\mathcal{M}_{\leq r}$ be a descent direction. Then there exists an L_k that satisfies the condition in (17).*

Proof. Since $\boldsymbol{\zeta}_k$ is a descent direction, it follows that $\mathbf{0} \notin \text{grad}f(\mathbf{X}_k) + \partial\|\mathbf{X}\|_*$ and $\langle \text{grad}f(\mathbf{X}_k), \boldsymbol{\zeta}_k \rangle < 0$. Since $T_L(\mathbf{X}_k)$ is continuous in L , there must exist an \hat{L} such that $\Psi(T_L(\mathbf{X}_k)) \leq \Psi(\mathbf{X}_k) + \beta \langle \text{grad}f(\mathbf{X}_k), \boldsymbol{\zeta}_k \rangle / L$, $\forall L \in [\hat{L}, +\infty)$. \square

In general, optimization methods on Riemannian manifolds are guaranteed to be locally convergent, and it is nontrivial to check whether a limit point \mathbf{X}^* is a global solution or not. However, for PRG, the limit point \mathbf{X}^* will be a global solution if $r > \text{rank}(\mathbf{X}^*)$.

Theorem 1. *Let $\{\mathbf{X}_k\}$ be an infinite sequence of iterates generated by Algorithm 1. Then every accumulation point of $\{\mathbf{X}_k\}$ is a critical point of f over $\mathcal{M}_{\leq r}$. Furthermore, $\lim_{k \rightarrow \infty} \|\text{grad}f(\mathbf{X}_k) + \boldsymbol{\zeta}\|_F = 0$. Let \mathbf{X}^* denote the limit point. In particular, if $\text{rank}(\mathbf{X}^*) < r$, then we have $\nabla f(\mathbf{X}^*) + \boldsymbol{\zeta} = \mathbf{0}$, i.e., \mathbf{X}^* is a global optimum to (14).*

Proof. Note that $\Psi(\mathbf{X})$ is bounded below. The proof can be completed by adapting the proof of Theorem 3.9 in [38]. \square

Stopping conditions of PRG. For simplicity, we stop PRG if the following condition is achieved:

$$\frac{\Psi(\mathbf{X}_{k-1}) - \Psi(\mathbf{X}_k)}{\Psi(\mathbf{X}_{k-1})} \leq \epsilon, \quad (19)$$

where ϵ denotes a tolerance value.

3.2 Robust PRG on $\mathcal{M}_{\leq r}$ with Outliers

Now, we extend PRG to minimize $\Psi(\mathbf{X}, \mathbf{E})$ in (12) regarding the outlier cases. For convenience, define

$$f(\mathbf{X}, \mathbf{E}) = \frac{\gamma}{2} \|\mathcal{A}(\mathbf{X}) + \mathcal{B}(\mathbf{E}) - \mathbf{D}\|_F^2.$$

We then need to solve the following problem:

$$\min_{\mathbf{X} \in \mathcal{M}_{\leq r}, \mathbf{E}} \|\mathbf{X}\|_* + \lambda \Upsilon(\mathbf{E}) + f(\mathbf{X}, \mathbf{E}). \quad (20)$$

Following [25], we optimize the two variables \mathbf{X} and \mathbf{E} using an alternating approach. Let the pair $(\mathbf{X}_k, \mathbf{E}_k)$ denote

Table 1: Computation of $S_\lambda(\mathbf{B})$.

| $\Upsilon(\mathbf{E})$ | MR: $\ \mathbf{E}\ _1$ | LRR: $\ \mathbf{E}\ _{2,1}$ |
|-------------------------|---|---|
| $S_\lambda(\mathbf{B})$ | $\text{sgn}(\mathbf{B}) \odot \max(\mathbf{B} - \frac{\lambda}{\gamma}, 0)$ | $[S_\lambda(\mathbf{B})]_i = \frac{\max(\ \mathbf{b}_i\ - \frac{\lambda}{\gamma}, 0)}{\ \mathbf{b}_i\ } \mathbf{b}_i, \forall i$ |

the variables obtained from the k -iteration. At the $(k+1)$ th iteration, we update \mathbf{X} and \mathbf{E} as below:

To update \mathbf{X} , we fix $\mathbf{E} = \mathbf{E}_k$ and minimize a local model of $\Psi(\mathbf{X}, \mathbf{E})$ w.r.t. \mathbf{X} :

$$m_L(\mathbf{X}; \mathbf{X}_k, \mathbf{E}_k) := \|\mathbf{X}\|_* + f(\mathbf{X}_k, \mathbf{E}_k) + \langle \text{grad} f(\mathbf{X}_k, \mathbf{E}_k), \xi \rangle + L/2 \langle \xi, \xi \rangle,$$

where $\mathbf{X} = \mathbf{X}_k + \xi$, $\xi \in T_{\mathbf{X}_k} \mathcal{M}_{\leq r}$, and L is a positive number. Let $T_L(\mathbf{X}_k, \mathbf{E}_k)$ denote the minimizer of $m_L(\mathbf{X}; \mathbf{X}_k, \mathbf{E}_k)$. Then $T_L(\mathbf{X}_k, \mathbf{E}_k)$ can be computed according to Lemma 1, where L is determined by Armijo line search to make a sufficient decrease of the objective.

To update \mathbf{E} , we fix $\mathbf{X} = \mathbf{X}_{k+1}$ and solve a problem:

$$\min_{\mathbf{E}} \lambda \Upsilon(\mathbf{E}) + \frac{\gamma}{2} \|\mathcal{A}(\mathbf{X}_{k+1}) + \mathcal{B}(\mathbf{E}) - \mathbf{D}\|_F^2. \quad (21)$$

Solving this problem with general \mathcal{B} would be very difficult. However, for MR and LRR, $\mathcal{B}(\mathbf{E}) = \mathbf{E}$ and $\Upsilon(\mathbf{E})$ is either $\|\mathbf{E}\|_1$ or $\|\mathbf{E}\|_{2,1}$. As a result, the problem (21) has a closed-form solution. Let us define $\mathbf{B}_k = \mathbf{D} - \mathcal{A}(\mathbf{X}_{k+1})$. Then \mathbf{B}_k is a vector for MR and a matrix in the form of $\mathbf{B}_k = [\mathbf{b}_1^k, \dots, \mathbf{b}_n^k]$ for LRR. The closed-form solution, denoted by $S_\lambda(\mathbf{B}_k)$, is shown in Table 1. In cases where the problem (21) cannot be solved in closed-form, one may adopt iterative procedures to solve it.

The detailed algorithm, which is referred to as robust PRG (RPRG), is shown in Algorithm 2. Due to the possible ill-conditioned issues,⁴ we apply a homotopy continuation technique to accelerate the convergence speed. Starting from an initial guess λ_0 , we set $\lambda_k = \min(\lambda_0 \rho^{k-1}, \lambda)$ and compute $\mathbf{E}_k = S_{\lambda_k}(\mathbf{X}_k, \mathbf{E}_{k-1})$, where ρ is chosen from $(0, 1)$. Clearly, λ_k is non-increasing w.r.t. k .

Algorithm 2 Robust PRG for Solving Problem (11).

Require: Initial $(\mathbf{X}_0, \mathbf{E}_0)$, parameter λ and γ , initial λ_0 , parameter $r, \rho \in (0, 1)$, stopping tolerance ϵ .

- 1: For $k = 1, \dots, K$
 - 2: Let $\lambda_k = \max(\lambda_0 \rho^{k-1}, \lambda)$.
 - 3: Compute $\text{grad} f(\mathbf{X}_{k-1}, \mathbf{E}_{k-1})$ by (5) or (9).
 - 4: Choose L_k by Armijo line search. Set $\mathbf{X}_k = T_{L_k}(\mathbf{X}_{k-1}, \mathbf{E}_{k-1})$.
 - 5: Compute $\mathbf{E}_k = S_{\lambda_k}(\mathbf{B}_{k-1})$, where $\mathbf{B}_{k-1} = \mathbf{D} - \mathcal{A}(\mathbf{X}_k)$.
 - 6: Terminate if stopping conditions are achieved.
 - 7: End
 - 8: Return $(\mathbf{X}_k, \mathbf{E}_k)$.
-

⁴When λ is very small or γ is very large, $\|\mathbf{E}\|_1$ can be very large at the beginning due to the thresholding in Table 1, making \mathbf{X}_1 far from its optimum.

We discuss the convergence as follows.

Proposition 1. Let $\Psi(\mathbf{X}_k, \mathbf{E}_k) = \|\mathbf{X}\|_* + \lambda_k \Upsilon(\mathbf{E}) + f(\mathbf{X}, \mathbf{E})$, and $\{(\mathbf{X}_k, \mathbf{E}_k)\}$ be an infinite sequence of iterates generated by Algorithm 2. It follows that $\Psi(\mathbf{X}_{k+1}, \mathbf{E}_{k+1}) \leq \Psi(\mathbf{X}_k, \mathbf{E}_k)$, and $\{(\mathbf{X}_k, \mathbf{E}_k)\}$ converges to a limit point $(\mathbf{X}^*, \mathbf{E}^*)$.

Please find the proof in supplementary file. Thee stopping condition in (19) can be extended to RPRG by replacing $\Psi(\mathbf{X})$ with $\Psi(\mathbf{X}, \mathbf{E})$.

4 Subspace Pursuit for Solving Problem (1)

Both PRG and RPRG methods operate on $\mathcal{M}_{\leq r}$, which rely on the knowledge of r . Unfortunately, the parameter r is usually unknown. Based on Theorem 1, one should set a sufficiently large r such that $r \geq \text{rank}(\mathbf{X}^*)$, where \mathbf{X}^* is an optimal solution of problem (1). However, the computational cost will dramatically increase if r is too large.

Regarding the above issues, we propose a subspace pursuit (SP) paradigm to address problem (1). To introduce SP, we bring in an additional integer κ which is **assumed to be several times smaller** than $\text{rank}(\mathbf{X}^*)$. Taking the outlier cases for example, instead of doing RPRG with a large r , we gradually increase the rank of \mathbf{X} from $\text{rank}(\mathbf{X}) = 0$ (e.g., $\mathbf{X} = \mathbf{0}$) by a small value κ , and perform RPRG to the following subproblem with increased t .

$$\min_{\mathbf{X}, \mathbf{E}} \Psi(\mathbf{X}, \mathbf{E}), \quad \text{s.t. } \text{rank}(\mathbf{X}) \leq t\kappa, \quad (22)$$

where t denotes the iteration index. Essentially, the subspace pursuit addresses problem (1) by solving a series of subproblems in (22) using RPRG on $\mathcal{M}_{\leq t\kappa}$ (see Step 8 of Algorithm 3), where $t = 1, \dots, T$ and T denotes the maximum number of iterations.

Algorithm 3 Subspace Pursuit for Solving (1).

Require: Parameters κ, λ and γ , initial λ_0, χ, ρ (where $\chi < \rho$), and stopping tolerance ϵ and ϵ .

- 1: Initialize $\mathbf{X}^0 = \mathbf{0}, \mathbf{E}^0 = \mathbf{0}$, and $\lambda_0^0 = \lambda_0$.
 - 2: For $t = 1, \dots, T$
 - 3: Let $r = t\kappa$ and $s = (t-1)\kappa$.
 - 4: Let $\lambda_0^t = \lambda^{t-1}$, and $\lambda^t = \max(\lambda_0 \chi^t, \lambda)$.
 - 5: Compute $\mathbf{G} = \gamma \mathcal{A}^*(\mathcal{A}(\mathbf{X}^{t-1}) + \mathcal{B}(\mathbf{E}^{t-1}) - \mathbf{D})$.
 - 6: Compute $\text{grad} f(\mathbf{X}^{t-1}, \mathbf{E}^{t-1}) = P_{T_{\mathbf{X}^{t-1}} \mathcal{M}_s}(\mathbf{G}) + \Xi_\kappa^{t-1}$.
 - 7: Choose L_t to satisfy (17). Set $\mathbf{X}_0^t = T_{L_t}(\mathbf{X}^{t-1}, \mathbf{E}^{t-1})$ and $\mathbf{E}_0^t = S_{\lambda_0^t}(\mathbf{B}^{t-1})$, where $\mathbf{B}^{t-1} = \mathbf{D} - \mathcal{A}(\mathbf{X}^t)$.
 - 8: Update $(\mathbf{X}^t, \mathbf{E}^t)$ by calling RPRG to address (22) with initial input $(\mathbf{X}_0^t, \mathbf{E}_0^t)$, $r, \lambda_0^t, \lambda^t$, and ϵ .
 - 9: Terminate if stopping conditions are achieved.
 - 10: End
-

At the t th iteration, SP either stops or increase r from $t\kappa$ to $(t+1)\kappa$ and the domain of \mathbf{X} changes from a Riemannian manifold $\mathcal{M}_{t\kappa}$ to $\mathcal{M}_{\leq (t+1)\kappa}$. As a result, we need to

compute Ξ_{r-s} in Step 6 to calculate $\text{grad}f(\mathbf{X}^{t-1}, \mathbf{E}^{t-1})$ according to equation (9).

To accelerate the convergence speed, two techniques are critical, i.e. the warm start and homotopy continuation techniques. To warm start, we prepare a good initial guess of $(\mathbf{X}_0^t, \mathbf{E}_0^t)$ for RPRG in Steps 4-6 (Steps 4-6 are exactly Steps 4-5 in Algorithm 2). To apply the continuation technique in RPRG. To facilitate its usage in SP, in Step 8, we adaptively set the initial λ_0 and target λ for RPRG by λ_0^t and λ^t (see Step 4), respectively. Note the parameter χ should be smaller than ρ in RPRG.

Note that PRG can be also incorporated into the SP framework. For convenience, we refer SP with RPRG and PRG to as SP-RPRG and SP-PRG, respectively.

SP increase the rank by κ iteratively. Due to limited size of \mathbf{X} , SP will be stopped in limited steps. Moreover, the objective value $\Psi(\mathbf{X}, \mathbf{E})$ monotonically decrease w.r.t. t .

Proposition 2. *Let $\{\mathbf{X}^t, \mathbf{E}^t\}$ be the sequence generated by Algorithm 3. Then we have*

$$\Psi(\mathbf{X}^t, \mathbf{E}^t) \leq \Psi(\mathbf{X}^{t-1}, \mathbf{E}^{t-1}) - \beta \|\Xi_{\kappa}^{t-1}\|_F^2 / L_t. \quad (23)$$

Please find the proof in supplementary file.

Stopping conditions of SP. Let \mathbf{X}_t^* be the limit point of RPRG at the t th iteration. According to Theorem 1, if $\text{rank}(\mathbf{X}_t^*) < t\kappa$, \mathbf{X}_t^* must be a global solution. As a result, SP will stop at the t th iteration. According to Proposition 2, if $\|\Xi_{\kappa}^{t-1}\|_F^2$ is very small, then there is no need to proceed. Then we may also stop the iterations if the following condition is achieved:

$$\frac{\Psi(\mathbf{X}^{t-1}, \mathbf{E}^{t-1}) - \Psi(\mathbf{X}^t, \mathbf{E}^t)}{\kappa \Psi(\mathbf{X}^{t-1}, \mathbf{E}^{t-1})} \leq \varepsilon, \quad (24)$$

where ε denotes a tolerance value.

4.1 Parameter settings

For convenience of parameter setting, we suggest choosing the penalty parameter γ in (22) according to $\gamma = 1/(\nu\sigma_1)$, where ν is a scaling factor, and σ denotes the singular vector of $\mathcal{A}^*(\mathbf{D})$.⁵ For robust cases, the parameter λ in (12) is chosen by $\lambda = \delta\gamma d_m$, where d_m denotes the mean of $|\mathbf{D}|$. The integer κ is chosen such that $\sigma_i \geq \eta\sigma_1$, $\forall i \leq \kappa$, and $\sigma_{\kappa+1} < \eta\sigma_1$. Without loss of generality, we suggest setting $\nu \in (0.0001, 0.01)$, $\delta \in (0.01, 1]$, and $\eta \in (0.5, 0.9)$. One may also apply cross-validations to choose ν and δ regarding model parameters. Lastly, in SP, we do not need to optimize each subproblem accurately. Therefore, in SP we suggest setting $\epsilon = 0.01$ for PRG and RPRG.

⁵The setting of γ is consistent with the setting of μ in *matrix lasso* in [43], where $\gamma = 1$, and $\mu = \nu\sigma_1$ is suggested in general.

4.2 Complexity Analysis

The complexity of SP mainly includes two parts, i.e., the computation of Ξ_{κ}^t which can be done by truncated SVD of rank κ and the subproblem optimization by PRG or RPRG.

Here, we focus on the complexity of SP on MR. At the t th iteration of SP, the complexity of PRG or RPRG is $O((m+n)(t\kappa)^2 + lt\kappa)$, where $t\kappa \leq r + \kappa$. For sufficiently sparse matrices like in MR, the truncated SVD of rank κ in SP can be completed in $O((m+n)\kappa)$ using PROPACK [22]; while the truncated SVD in existing proximal gradient based methods takes $O((m+n)r)$, where κ is several times smaller than r . Note that in general SP needs only $\lceil r/\kappa \rceil$ times of truncated SVDs. Therefore, SP is cheaper than existing proximal gradient based methods on MR. The complexity comparison on LRR can be found in supplementary file.

5 Related Work

The authors in [31] exploited Riemannian structures and presented a trust-region algorithm to address trace-norm minimizations. The proposed method, denoted by MMBS, alternates between fixed-rank optimization and rank-one updates. However, this method shows slower speed than APG on large-scale problems [31]. The authors in [28] proposed a Grassmannian manifold method to address trace-norm minimizations on a fixed-rank manifold. In general, this method has similar complexity to ScGrassMC that also operates on Grassmannian manifold [33]

Active subspace methods or greedy methods, that increase the rank by one per iteration, have gained great attention in recent years [15, 18, 40, 23]. However, these methods usually involve expensive subproblems, and might be very expensive when the true rank is high. For example, Laue's method [23] needs to solve nonlinear master problems using the BFGS method, which can be very expensive for large-scale problems. More recently, [17] proposed a novel active subspace selection method for solving trace-norm regularized problems. However, this method may suffer from slow convergence speed due to the approximated SVDs and inefficient solvers for the subproblem optimization. In [42], the authors proposed a Riemannian pursuit (RP) algorithm which increases the rank more than one. However, this algorithm cannot deal with trace-norm regularized problems.

6 Experimental Results

We evaluate the proposed methods on for two classical trace-norm based tasks, namely low-rank matrix completion and LRR based clustering. All the experiments are conducted in Matlab (R2012b) on a PC installed a 64-bit operating system with an Intel(R) Core(TM) i7 CPU

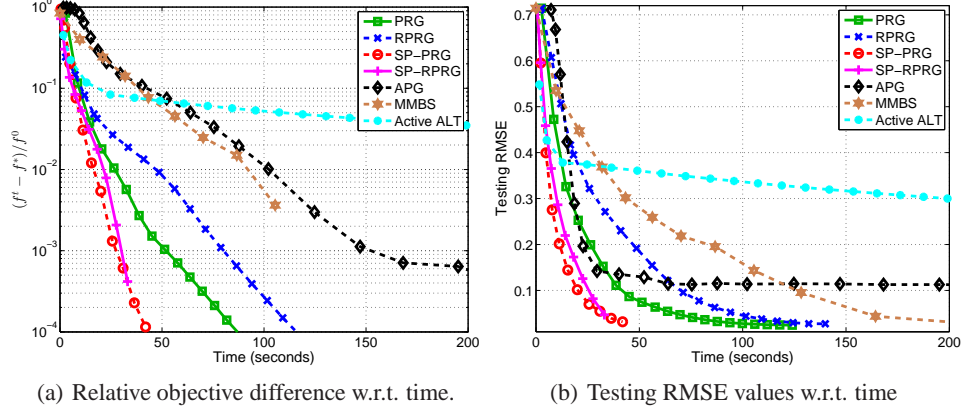


Figure 1: Performance of various methods on **TOY1**.

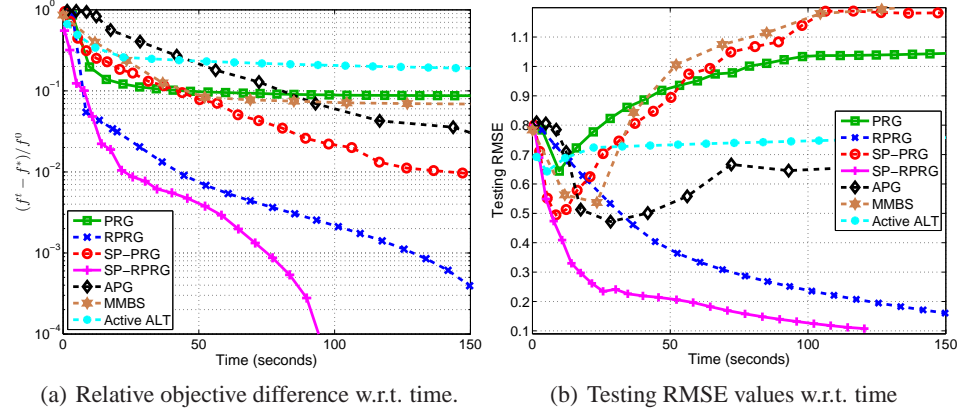


Figure 2: Performance of various methods on **TOY2**, where 5% of data are disturbed by severe outliers. While other methods over-fit on this data, the proposed robust PRG (RPRG) and SP-RPRG still achieve promising testing RMSE.

(3.2GHz with single-thread mode) and 64GB memory.

6.1 Experiments on Matrix Completion

We study the performance of proposed methods, namely PRG, RPRG, SP-PRG and SP-RPRG, on the matrix completion task. Three state-of-the-art trace-norm based methods, e.g. APG [43], MMBS [31], and Active ALT [17], are adopted as baselines. Moreover, to study the efficiency of proposed methods on matrix completion, we also compare several efficient fixed-rank methods, including RP [42], LMaFit [47], ScGrassMC [33], LRGeomCG [44], qGeomMC [30].⁶ We do not report the results of some methods, such as IALM [24] and method in [28], since they are either slower than the compared methods in this paper or

the source codes are not available.

We adopt the root-mean-square error (RMSE) testing set as a major evaluation metric: $\text{RMSE} = \|\mathcal{P}_\Omega(\mathbf{D} - \mathbf{X}^*)\|_F / \sqrt{(|\Omega|)}$, where \mathbf{X}^* denotes the recovered matrix, and Ω denotes the index set of a testing set, and \mathcal{P}_Ω denotes the orthogonal projection onto Ω [44].

6.1.1 Synthetic Experiments

Following [33, 42], we generate ground-truth low-rank matrices $\mathbf{D} = \mathbf{U}\text{diag}(\boldsymbol{\sigma})\mathbf{V}^\top \in \mathbb{R}^{m \times m}$ of rank r , where $\mathbf{U} \in \text{St}_r^m, \mathbf{V} \in \text{St}_r^m, m = 5000, r = 50$ and $\boldsymbol{\sigma}$ is a 50-dimensional vector with its entries sampled from a uniform distribution $[0, 1000]$. We sample $l = \omega r(2m - r)$ entries from \mathbf{D} uniformly as the observations stored in $\mathbf{d} \in \mathbb{R}^l$, where ω is an oversampling factor [24]. Here, we set $\omega = 2.5$. We study two toy data sets whose observations are perturbed by two kind of noises: In the first toy data set **TOY1**, each entry of \mathbf{d} is perturbed by additive Gaussian noise of magnitude $0.01\|\mathbf{d}\|_2/\|\mathbf{n}\|_2$, where $\mathbf{n} \in \mathbb{R}^l$ is a Gaussian vector with each entry being sam-

⁶APG is available from <http://www.math.nus.edu.sg/~mctc/alg/apg/>, RP is available from <http://www.tanmingkui.com/rp.html>, Active ALT is available from <http://www.cs.utexas.edu/~cjhsieh/>, MMBS, LMaFit, ScGrassMC, qGeomMC and LRGeomCG are available from: <http://www.montefiore.ulg.ac.be/~mishra/fixedrank/>.

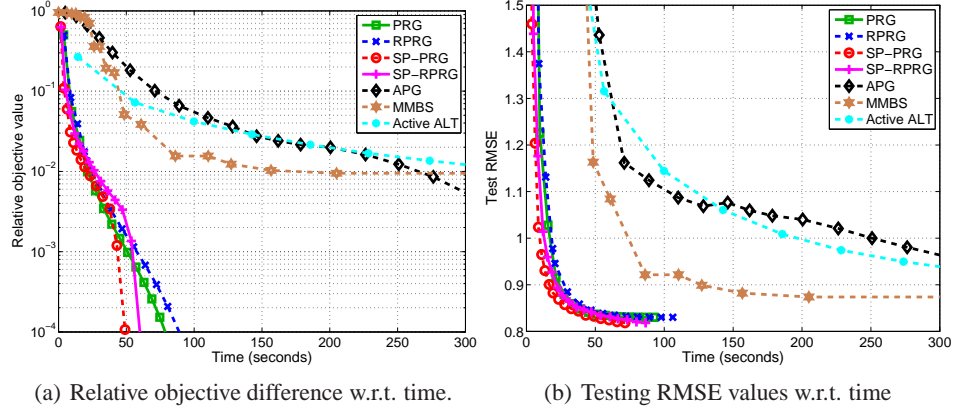


Figure 3: Performance of various methods on Movie-10M data set.

pled from $N(0, 1)$; The second toy data **TOY2** is obtained based on **TOY1**, by further perturbing 5% of the observations with outliers uniformly sampled from $[-10, 10]$. In these synthetic experiments, we set $\nu = 0.005$ $\eta = 0.65$, and $\delta = 0.1$.

Three trace-norm based methods APG, MMBS and Active ALT, are adopted as the baselines. The *Relative objective difference* and *Testing RMSE* w.r.t. time on **TOY1** and **TOY2** are reported in Figures 1 and 2, respectively.

According to Figure 1(a), our proposed PRG, RPRG, SP-PRG and SP-RPRG converge much faster than the comparators, and SP-PRG and SP-RPRG improve upon their counterparts (*i.e.*, PRG and RPRG) significantly. From Figure 1(b), the testing RMSE shows similar trends to the objective values. Note that, our methods thus achieve low RMSE values in very short times. In general, the Active ALT method is slower than others, which may be due to the approximated SVDs and inefficient solvers for the subproblem optimization.

From Figure 2(a), on **TOY2** which is disturbed by outliers, our proposed method in general converges faster than the baselines. However, from Figure 2(b), only the proposed RPRG and SP-RPRG achieve promising testing RMSE values. While other methods over-fit the data after several iterations due to the outliers. Not also that SP-RPRG converges faster than its counterpart RPRG. This observation demonstrates the effectiveness and efficiency of our proposed methods.

6.1.2 Experiments on Real-world Data

We study the performance of SP-PRG and SR-PRG on three collaborative filtering data sets: MovieLens with 10M ratings (denoted by Movie-10M) [16], Netflix Prize dataset [20] and Yahoo! Music Track 1 data set [13]. The statistics of these data sets are recorded in Table 2.

In the first experiment, we only compare with the three

| Data set | m | n | $ \Omega $ |
|-----------|-----------|---------|-------------|
| Movie-10M | 71,567 | 10,677 | 10,000,054 |
| Netflix | 48,089 | 17,770 | 100,480,507 |
| Yahoo | 1,000,990 | 624,961 | 252,800,275 |

trace-norm based methods, *e.g.* APG [43], MMBS [31] and Active ALT [17] on Movie-10M. We report the change of *Relative objective difference* and *Testing RMSE* w.r.t. time in Figure 3. Here, we randomly choose 80% of the ratings as training set and the remainder as the testing set. From Figures 3(a) and 3(b), our proposed methods show much faster convergence speed as well as faster decreasing of testing RMSE values.

In the second experiment, the baseline methods include APG [43], MMBS [31], LRGeomCG [44], qGeomMC [30], Lmafit [47], Active ALT [17], ScGrassMC [33] and RP [42]. The ranks returned by SP-PRG are used as the rank estimations for fixed-rank methods, such as ScGrassMC, qGeomMC and Lmafit. We set $\nu = 0.001$ $\eta = 0.65$, and $\delta = 0.7$. Following [23, 40, 18], we report the testing RMSE of different methods over 10 random 80/20 training/testing partitions.

Comparison results are shown in Table 3. Note that we did not obtain the results of some methods on two larger data sets, *i.e.* Netflix and Yahoo, due to the very expensive computation cost. We thus leave the results blank. According to the table, the proposed SP-PRG and SP-RPRG methods achieve better testing RMSE than RP with comparable time. Note that RP relies on carefully designed stopping conditions to induce low-rank solutions, and cannot deal with outliers [42]. In particular, SP-PRG and SP-RPRG achieve significant improvements in terms of testing RMSE on the Yahoo data set.

Table 3: Experimental results on real-world datasets, where time is recorded in seconds.

| Method | Movie-10M | | Netflix | | Yahoo | |
|-----------|-----------|--------|---------|---------|--------|-------|
| | RMSE | Time | RMSE | Time | RMSE | Time |
| APG | 1.094 | 810.01 | 1.038 | 2883.80 | — | — |
| LRGeomCG | 0.823 | 57.67 | 0.860 | 2356.86 | 25.228 | 18319 |
| QgeomMC | 0.836 | 96.41 | 0.897 | 9794.75 | 24.167 | 82419 |
| Lmafit | 0.838 | 133.86 | 0.876 | 2683.73 | 24.368 | 24349 |
| ALT | 0.855 | 917.17 | — | — | — | — |
| MMBS | 0.821 | 441.10 | — | — | — | — |
| ScGrassMC | 0.845 | 216.07 | 0.892 | 4522.68 | 24.954 | 37705 |
| RP | 0.818 | 46.56 | 0.858 | 1143.02 | 23.451 | 12456 |
| SP-PRG | 0.817 | 53.42 | 0.855 | 1057.35 | 22.644 | 15972 |
| SP-RPRG | 0.815 | 67.73 | 0.857 | 1245.15 | 22.537 | 17263 |

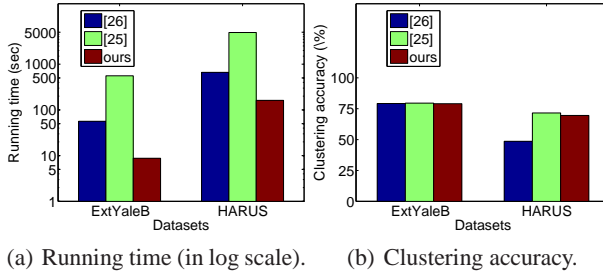


Figure 4: Comparing the running times and clustering accuracies of different LRR solvers on two datasets.

6.2 Experiments on LRR Based Subspace Clustering

To compare our proposed SP-RPRG method with the existing LRR solvers in [25, 26], we conduct experiments on the Extended Yale Face Database B (**ExtYaleB**) for face clustering, and the Human Activity Recognition Using Smartphones dataset (**HARUS**) [3] for human activity clustering. Following [26], the clustering performance is measured by *clustering accuracy*, namely the number of correctly clustered samples over the total number of samples.

The ExtYaleB dataset contains 2,414 frontal face images of 38 subjects with different lighting, poses and illumination conditions, where each subject has round 64 faces. Following [27], we use 640 faces from the first 10 subjects. Each face image is resized to 48×42 pixels and then reshaped as a 2016-dimensional gray-level intensity feature. The HARUS dataset is a large dataset (containing 10,299 signals w.r.t. 6 activities) with data collected using embedded sensors on the smartphones carried by volunteers on their waists, when they are conducting daily activities (e.g., walking, sitting, laying). The captured sensor signals are pre-processed to filter noise and post-processed. Finally, a 561-dimensional feature vector with time and frequency domain variables is extracted for each signal.

The best clustering accuracies and the corresponding running times are reported in Figure 4(b) and Figure 4(a), respectively. It can be observed that, our SP-RPRG method outperforms the two existing LRR solvers in terms of efficiency, since our algorithm does not frequently involve

SVDs w.r.t. large matrices. Moreover, our algorithm achieves comparable clustering performance with [25]. In contrast, the LRR solver in [26] achieves lower clustering accuracy on the HARUS dataset, possibly because that algorithm is not guaranteed to obtain a globally optimal solution.

7 Conclusion

Classical proximal methods may require many large-rank SVDs when addressing the trace-norm regularized problems on vector spaces. To overcome this, we first propose a proximal Riemannian gradient (PRG) method to address trace-norm regularized problems over a matrix variety $\mathcal{M}_{\leq r}$, where r is supposed to be known. By performing optimization on $\mathcal{M}_{\leq r}$, PRG does not require SVDs, thus can greatly reduce the computation cost. A robust version of PRG method has also been proposed to handle the outlier cases. To address general trace-norm regularized problems, a subspace pursuit strategy is proposed by iteratively activating a number of active subspaces. Extensive experiments on two classical trace-norm based tasks, namely low-rank matrix completion and LRR based clustering, demonstrate the superior efficiency of the proposed methods over other methods.

References

- [1] P-A Absil, L. Amodei, and G. Meyer. Two Newton methods on the manifold of fixed-rank matrices endowed with Riemannian quotient geometries. *Comput. Stat.*, 29(3-4):569–590, 2014.
- [2] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2008.
- [3] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz. Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In *Ambient assisted living and home care*, pages 216–223. 2012.
- [4] A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Mach. Learn.*, 73(3):243–272, 2008.
- [5] H. Avron, S. Kale, S. Kasiviswanathan, and V. Sindhvani. Efficient and practical stochastic subgradient descent for nuclear norm regularization. Technical report, 2012.
- [6] N. Boumal and P.-A. Absil. RTRMC: A Riemannian trust-region method for low-rank matrix completion. In *NIPS*, 2012.
- [7] J. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM J. on Optim.*, 20(4):1956–1982, 2010.
- [8] E. J. Candès and Y. Plan. Tight oracle bounds for low-rank matrix recovery from a minimal number of random measurements. *IEEE Trans. Inform. Theory*, 57(4):2342–2359, 2010.

- [9] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Found. Comput. Math.*, 9:717–772, 2009.
- [10] Y. Chen, A. Jalali, S. Sanghavi, and C. Caramanis. Low-rank matrix recovery from errors and erasures. *IEEE Trans. Inform. Theory*, 59(7):4324–4337, 2013.
- [11] Y. Chen, H. Xu, C. Caramanis, and S. Sanghavi. Robust matrix completion with corrupted columns. In *ICML*, 2011.
- [12] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Model. Sim.*, 4(4):1168–1200, 2005.
- [13] G. Dror, N. Koenigstein, Y. Koren, and M. Weimer. The Yahoo! Music Dataset and KDD-Cup’11. *JMLR Workshop and Conference Proceedings*, 2012.
- [14] M. Fazel. Matrix rank minimization with applications. 2002. PhD thesis, Stanford University.
- [15] E. Hazan. Sparse approximate solutions to semidefinite programs. *LATIN*, pages 306–316, 2008.
- [16] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl. An algorithmic framework for performing collaborative filtering. In *SIGIR*, 1999.
- [17] C. Hsieh and P. Olsen. Nuclear norm minimization via active subspace selection. In *ICML*, 2014.
- [18] M. Jaggi and M. Sulovsky. A simple algorithm for nuclear norm regularized problems. In *ICML*, 2010.
- [19] S. Ji and J. Ye. An accelerated gradient method for trace norm minimization. In *ICML*. ACM, 2009.
- [20] KDDCup. ACM SIGKDD and netflix. In *Proceedings of KDD Cup and Workshop*, 2007.
- [21] R. H Keshavan, A. Montanari, and S. Oh. Matrix completion from noisy entries. *JMLR*, 99:2057–2078, 2010.
- [22] R. M. Larsen. Propack—software for large and sparse svd calculations. 2004.
- [23] S. Laue. A hybrid algorithm for convex semidefinite optimization. In *ICML*, 2012.
- [24] Z. Lin, M. Chen, and Y. Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. Technical report, UIUC, 2010.
- [25] Z. Lin, R. Liu, and Z. Su. Linearized alternating direction method with adaptive penalty for low-rank representation. *arXiv preprint arXiv:1109.0367*, 2011.
- [26] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Trans. Pattern Anal. Machine Intell.*, 35(1):171–184, 2013.
- [27] G. Liu, Z. Lin, and Y. Yu. Robust subspace segmentation by low-rank representation. In *ICML*, pages 663–670, 2010.
- [28] Y. Liu, F. Shang, H. Cheng, and J. Cheng. Nuclear norm regularized least squares optimization on grassmannian manifolds. In *UAI*, 2014.
- [29] G. Meyer, S. Bonnabel, and R. Sepulchre. Linear regression under fixed-rank constraints: A Riemannian approach. In *ICML*, 2011.
- [30] B. Mishra, K. A. Apuroop, and R. Sepulchre. A Riemannian geometry for low-rank matrix completion. Technical report, 2012.
- [31] B. Mishra, G. Meyer, F. Bach, and R. Sepulchre. Low-rank optimization with trace norm penalty. *SIAM J. Optim.*, 23(4):2124 – 2149, 2013.
- [32] Y. Nesterov. Gradient methods for minimizing composite objective function. Technical report, Center for Operations Research and Econometrics (CORE), Catholic University of Louvain (UCL), 2007.
- [33] T. T. Ngo and Y. Saad. Scaled gradients on Grassmann manifolds for matrix completion. In *NIPS*, 2012.
- [34] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma. Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images. *IEEE Trans. Pattern Anal. Machine Intell.*, 34(11):2233–2246, 2012.
- [35] B. Recht, M. Fazel, and P. A. Parrilo. Guaranteed minimum rank solutions to linear matrix equations via nuclear norm minimization. *SIAM Rev.*, 52(3), 2010.
- [36] B. Recht and C. Ré. Parallel stochastic gradient algorithms for large-scale matrix completion. *Math. Programm. Comput.*, 5(2):201–226, 2013.
- [37] R. Salakhutdinov and N. Srebro. Collaborative filtering in a non-uniform world: Learning with the weighted trace norm. In *NIPS*, 2010.
- [38] R. Schneider and A. Uschmajew. Convergence results for projected line-search methods on varieties of low-rank matrices via Łojasiewicz inequality. *arXiv preprint arXiv:1402.5284*, 2014.
- [39] Y. Shen, Z. Wen, and Y. Zhang. Augmented lagrangian alternating direction method for matrix separation based on low-rank factorization. *Optim. Method. and Softw.*, (ahead-of-print):1–25, 2012.
- [40] S. S. Shwartz, A. Gonen, and O. Shamir. Large-scale convex minimization with a low-rank constraint. In *ICML*, 2011.
- [41] S. Sra, S. Nowozin, and S. J Wright. *Optimization for Machine Learning*. MIT Press, 2011.
- [42] M. Tan, I. W. Tsang, L. Wang, B. Vandereycken, and S. J. Pan. Riemannian pursuit for big matrix recovery. In *ICML*, pages 1539–1547, 2014.
- [43] K.-C. Toh and S. Yun. An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems. *Pac. J. Optim.*, 6:615–640, 2010.
- [44] B. Vandereycken. Low-rank matrix completion by Riemannian optimization. *SIAM J. Optim.*, 23(2):1214–1236, 2013.
- [45] R. Vidal. A tutorial on subspace clustering. *IEEE Signal Proc. Mag.*, 28(2):52–68, 2010.
- [46] X. Wang, Z. Zhang, Y. Ma, X. Bai, W. Liu, and Z. Tu. Robust subspace discovery via relaxed rank minimization. *Neural comput.*, 26(3):611–635, 2014.

- [47] Z. Wen, W. Yin, and Y. Zhang. Solving a low-rank factorization model for matrix completion by a non-linear successive over-relaxation algorithm. *Math. Program. Comput.*, 4(4):333–361, 2012.
- [48] L. Xiao and T. Zhang. A proximal-gradient homotopy method for the sparse least-squares problem. *SIAM J. Optim.*, 23(2):1062–1091, 2013.
- [49] M. Yuan, A. Ekici, Z. Lu, and R. Monteiro. Dimension reduction and coefficient estimation in multivariate linear regression. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 69(3):329–346, 2007.
- [50] X. Zhang, D. Schuurmans, and Y. Yu. Accelerated training for matrix-norm regularization: A boosting approach. In *NIPS*, pages 2906–2914, 2012.